AMG-12 RTI Update

Presented to

Architecture Management Group

AMG 12
Alexandria, VA
12-13 June 1996

James Calvin, MIT Lincoln Laboratory
Richard Weatherly, MITRE

jcalvin@ll.mit.edu
weather@mitre.org

Agenda

- RTI Status
- Update on deadlock avoidance in the RTI
- RTI Management activities
- CORBA 2.0 interoperability investigation

RTI Status

|	Release 0.1 Released 31 Oct 95	Release 0.2 Released 29 Feb 96	Release 0.3 Released 30 Apr 96	Release 0.3e Released 3 Jun 96
Federation Management	Create Federation Execution, Destroy Federation Execution, Join Federation Execution, Resign Federation Execution			
Declaration Management	Publish/Subscribe Object Class, Publish/Subscribe Interaction Class, Control Updates†, Control Interactions†			Filter space versions of subscribe/publish
Object Management	ID Request, Instantiate Object, Instantiate Discovered Object†, Delete Object, Remove Object†, Update Attribute Values, Reflect Attribute Values†, Send Interaction, Receive Interaction†		Cancel Object Reflection, Request Attribute Value Update, Provide Attribute Value Update†	
Ownership Management		All services described in the 0.3 specification		
Time Management	Time Advance Request, Next Event Request, Time Advance Grant†		Request Federate Time, Request Federation Rate, Set/Request Lookahead, Time Advance Request, Next Event Request, Time Advance Grant†	
Communications approach	All TCP/IP via ORBIX implementation; updates via central redistribution point	Initial multiple service model (TCP/IP, UDP, ...); reliable orb best effort transport	0.2 plus minimum rate (best effort variant) performance as req'd	As in 0.3
Latency scale	On the order of two TCP/IP xmit/recv times	Same as 0.1, except updates are on the order of a UDP xmit/recv time	As in 0.2, except additional services latency improved as required	
Filter support	Class and attribute name	as in 0.1	as in 0.1	0.1 plus initial data value filtering
# of federates supported	a few dozen	a few dozen	hundreds	
# of objects supported	a few dozen	a few hundred	~ thousand	

RTI Status (continued)

- Worked with protofederations with 0.3, 0.31, 0.32
- Will release 0.33 on 21 June 1996
- Planning has started for 1.0
- 0.3x series becomes a design reference for RTI 1.0
- By early July we expect to have a plan in place for 1.0 based on:
  - DMSO collected feedback for functionality and schedule requirements
  - Feedback from the prototype experience
 - Some experiments underway to support the planning process

What Was Done in Version 0.32 to Address Deadlock?

- Oneway services added to RTI_Ambassador interface
- Bug in a RTI_Ambassador call to the SIM_Ambassador caused it to wait forever
- RTI_Ambassador changed to favor calls from the Federate over calls to the SIM_Ambassador or from the rest of the RTI

Version 0.32 Process Structure

Federate

SIM Ambassador

RTI Ambassador

All Other Parts of the RTI
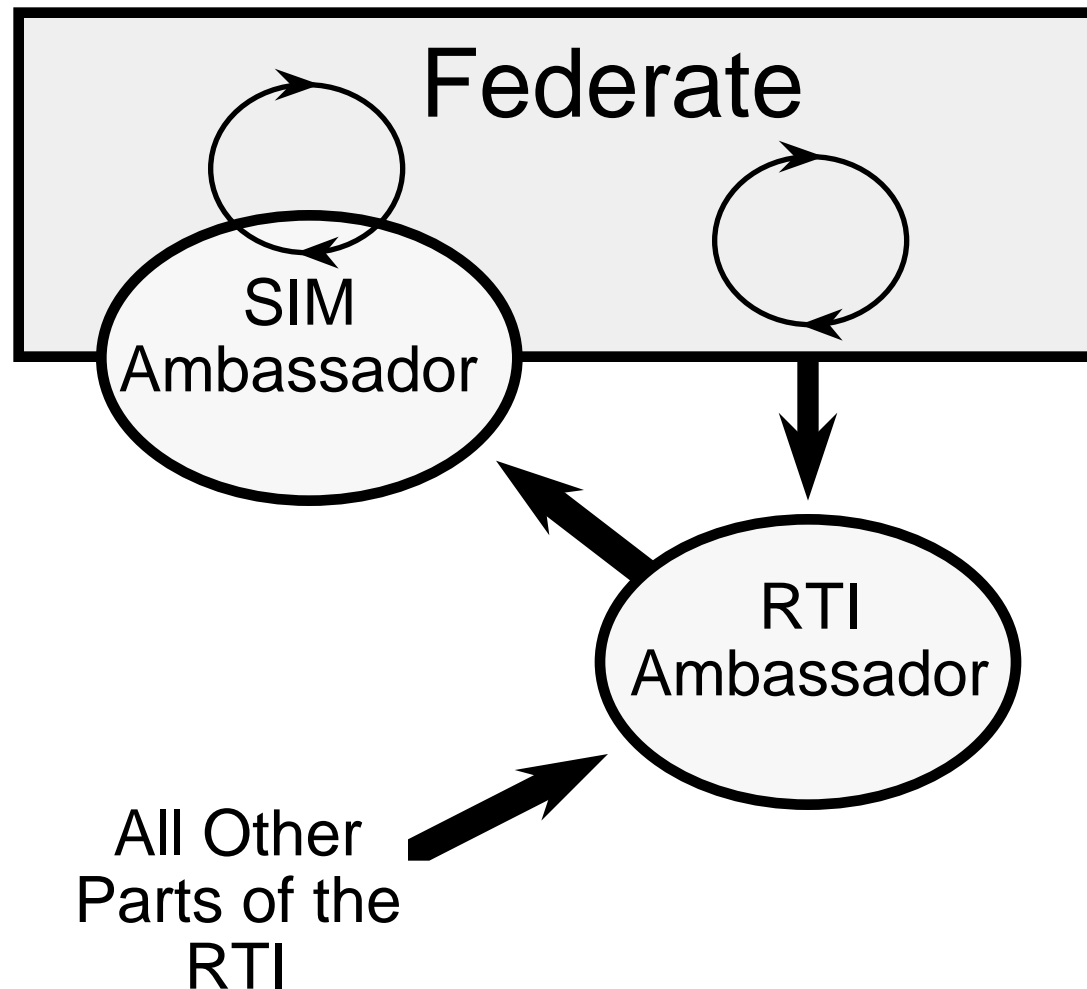
Version 0.32 with Single Thread Federate
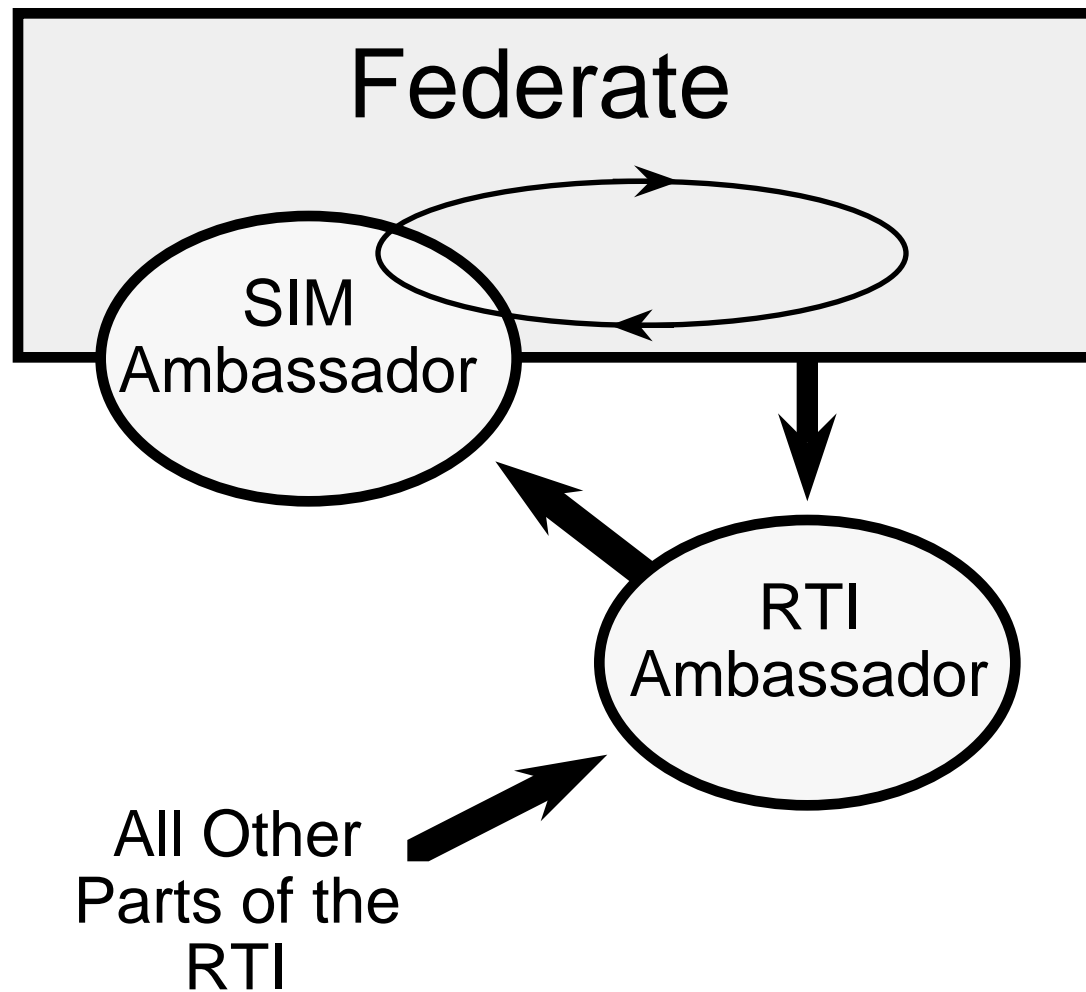
What Will Be Done in Version 0.33 to Address Deadlock?

- The Sim_Agent process/object introduced to decouple and queue calls from the RTI_Ambassador to the SIM_Ambassador

- Sequence numbers added to the calls from the RTI_Ambassador to the Sim_Agent to eliminate redundant federate information

- Oneway calls added to the Sim_Agent (former SIM_Ambassador interface)

- Split the reflect_attribute_values service in a way similar to how update_attribute_values was partitioned

Version 0.33 Process Structure
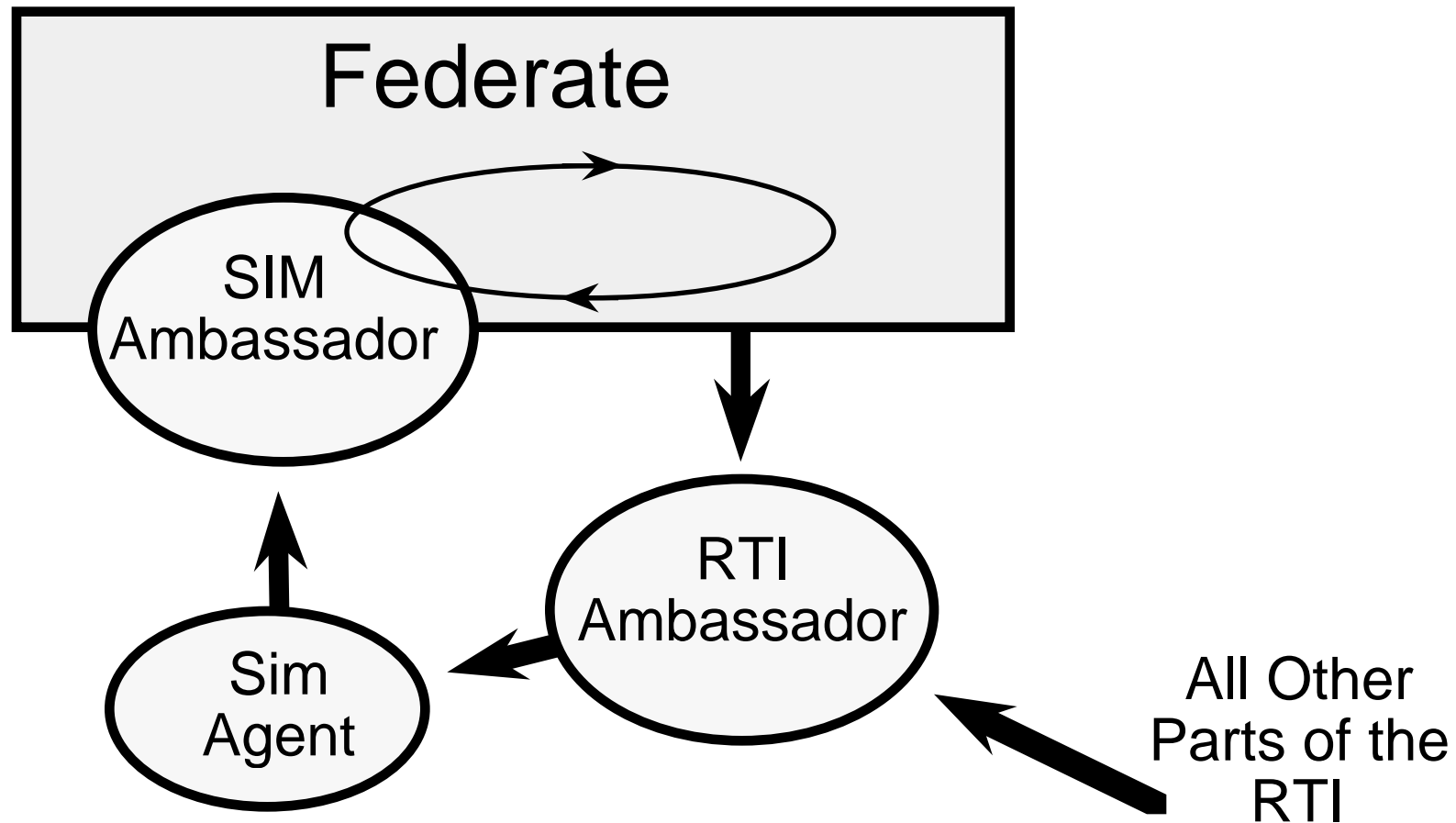
RTI Management Activities

- **Stood up in May 1996**
- **First meeting gathered ideas from individuals familiar with the conduct of actual military exercises**
- **"Buddy System" contact gave opportunity to observe protofederate management strategies**
- **Thesis: The RTI can be managed through its own subscription and object management services**
    - **No (or few) RTI management specific services will be needed**
    - **The RTI will have a SOM that is available to any federate**
    - **RTI management software will attach to the RTI like any normal federate, that is, join the federation to be managed**

CORBA 2.0 Interoperability Investigation

Various Vendor's Implementations of CORBA Should Operate Together

- CORBA 1.0 spec did not specify the format of data to be exchanged among objects
- Differences between vendor's implementations preclude their interoperability
- CORBA 2.0 spec (Aug 94) specifies an Internet Inter-Operability (IIOP) which all vendors must support to comply with CORBA 2.0

Compliant Implementations
Are Arriving

- **Several vendors have IIOP implementations now, including Iona (Orbix)**
- **All remaining major vendors have announced products within 12 months**
- **CORBA 2.0 compliance is considered a requirement for staying in the game**
- **Iona's initial release was faulty, as were others**
  - **They acknowledged the problems quickly**
  - **The latest release, 2.0.1, is intended chiefly to fix IIOP problems**

Testing and Assuring Interoperability Is Understood As a Community Problem

- Talks are underway at OMG about establishing a test suite and test activities
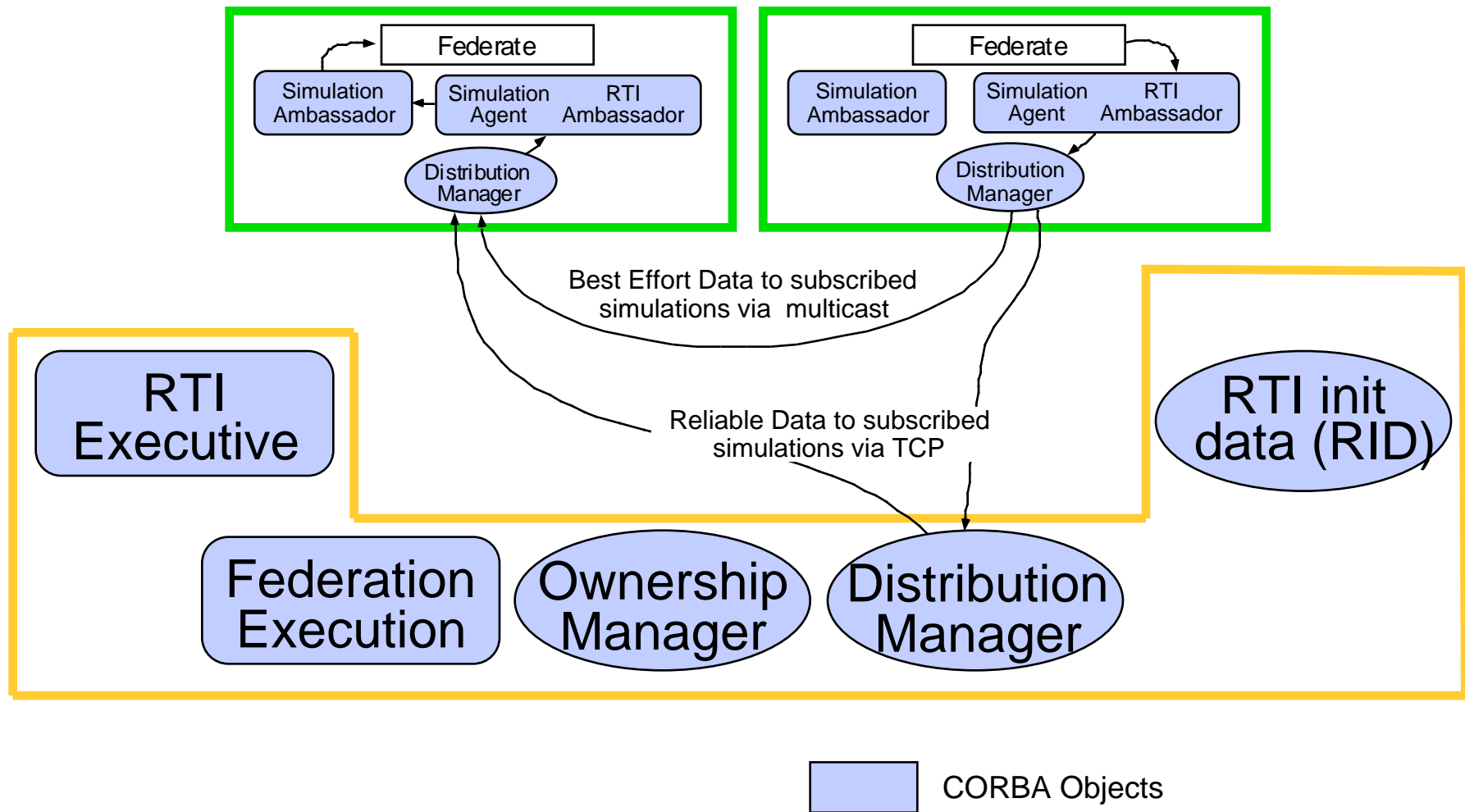- Vendors are funding a demonstration project at DSTC in Australia

RTI Prototype Is Undertaking a Limited Demonstration of Interoperability

- Project will implement a set of IDL interfaces representative of the most complicated features in the RTI

- The server side will be built in C++ using Orbix

- The client side will be built in Java using Black Widow from PostModern Computing

Runtime Infrastructure Version 0.3

Federate

Simulation Ambassador

Simulation Agent   RTI Ambassador

Distribution Manager

Federate

Simulation Ambassador

Simulation Agent   RTI Ambassador

Distribution Manager

Best Effort Data to subscribed simulations via multicast

Reliable Data to subscribed simulations via TCP

RTI Executive

RTI init data (RID)

Federation Execution

Ownership Manager

Distribution Manager

CORBA Objects

- Application programmer's interface is defined in OMG IDL
- Services are implemented (as far as user is concerned) as CORBA objects
- RTI prototype has been using Iona Orbix 1.3
- Expected advantages in managing distributed system:
  - Remote server activation on chosen host
  - Experiment with allocation of objects to servers
  - Independence of federate implementation language
- Known limitation: CORBA's comm characteristics and performance unsuitable for data comm services
  - Datagrams of few hundred bytes to be sent by thousands of players at 25 Hz with low latency
  - Exploit multicast for data segregation

Experience Has Been Mixed

- **CORBA interface to specialized comm protocols not as efficient as desired**

- **CORBA has worked well for management and control functions**

- **CORBA has worked well for centralized, reliable services**

- **Issues remain**
  - Efficiency of marshalling untyped data ('any')
    - Required to implement IDL interface to type-generic services
    - Implementation issue: Iona may provide help here
  - CORBA interoperability to support mixing vendors
  - Use of COTS layer across multiple platforms creates support and maintenance issue